

University of Groningen

Nested grids ILU-decomposition (NGILU)

Ploeg, A. van der; Botta, E.F.F.; Wubs, F.W.

Published in:
Journal of Computational and Applied Mathematics

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version
Publisher's PDF, also known as Version of record

Publication date:
1996

[Link to publication in University of Groningen/UMCG research database](#)

Citation for published version (APA):

Ploeg, A. V. D., Botta, E. F. F., & Wubs, F. W. (1996). Nested grids ILU-decomposition (NGILU). *Journal of Computational and Applied Mathematics*, 66(1-2), 515-526.

Copyright

Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

The publication may also be distributed here under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license. More information can be found on the University of Groningen website: <https://www.rug.nl/library/open-access/self-archiving-pure/taverne-amendment>.

Take-down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): <http://www.rug.nl/research/portal>. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.



Nested grids ILU-decomposition (NGILU)

A. van der Ploeg*, E.F.F. Botta, F.W. Wubs

Department of Mathematics, University of Groningen, P.O. Box 800, 9700 AV Groningen, The Netherlands

Received 31 August 1994; revised 18 May 1995

Abstract

A preconditioning technique is described which shows, in many cases, grid-independent convergence. This technique only requires an ordering of the unknowns based on the different levels of multigrid, and an incomplete LU-decomposition based on a drop tolerance. The method is demonstrated on a variety of well-known elliptic test problems including strongly varying coefficients, advective terms and grid refinement.

Keywords: Finite differences; Preconditioning; Multilevel methods; Conjugate gradients; Incomplete decomposition

AMS classification: 65F10; 65F50; 65N22; 65N55

1. Introduction

Solution of large sparse systems of linear equations continues to be a major research area with widespread application. In this paper, we consider linear systems of the form $Ax = b$, where A is a large $N \times N$ matrix and b a given vector. Elimination methods have as drawback that, in general, the number of nonzero entries strongly increases during the elimination process. Therefore, these methods are often too costly in terms of computer storage and CPU-time. As a consequence, sparse systems of linear equations are often solved by an iterative method, for example by some conjugate gradient-like method. In general, the speed of convergence of these methods strongly depends on the eigenvalue distribution of A . For example, for the conjugate gradient method (CG) it can be proven that it is very important that the spectral condition number is small, and that the extreme eigenvalues are well separated [17]. Therefore, a CG-like method is often applied to the preconditioned system $(LU)^{-1}Ax = (LU)^{-1}b$ instead of the original system $Ax = b$. Herein L and U are

* Corresponding author. E-mail: a.van.der.ploeg@math.rug.nl.

sparse factors obtained by some incomplete LU-decomposition of A . The matrix L is a lower-triangular matrix, and U is upper-triangular.

Meijerink and van der Vorst [10] have shown existence and uniqueness of the incomplete LU-decomposition for an important class of problems and for an arbitrary choice of the sparsity pattern of $L + U$. For symmetric systems, they showed that CG gives excellent results when combined with an incomplete Choleski-decomposition as preconditioner (ICCG). Gustafsson [8] has shown that, in many cases, the performance of an incomplete LU-decomposition can be improved by lumping all elements that are neglected during the decomposition on the main diagonal. For several problems, the CPU-time for the iteration combined with the resulting Modified ILU-decomposition (MILU) is $O(N^{5/4})$ in two dimensions and $O(N^{7/6})$ for 3D-problems, where N is the total number of unknowns (see also [2]).

Multigrid methods perform even better, and for a large class of problems they have an optimal order of convergence: the amount of work and storage is proportional to N . However, due to the required proper smoothers and the restriction and prolongation operators at each level, the implementation of multigrid techniques for practical problems is much more complicated than that of some CG-like method combined with a preconditioner. We present a method which combines the best properties of both, e.g. an incomplete LU-decomposition such that the preconditioned system can be solved with the optimal computational complexity $O(N)$ by some CG-like method. The basic idea is the same as in multigrid methods, which use coarser grids in order to remove the low-frequency errors effectively. Our preconditioning technique uses a partitioning of the unknowns based on the sequence of grids in multigrid.

In the special case when A comes from a standard discretisation of a stationary convection–diffusion equation on a uniform, rectangular grid, the partitioning of the unknowns is the same as in the so-called RRB-method described in [4], the nested recursive two-level decomposition method described in [1], and the method described in [11]. The lumping strategy in these methods is based on an approximation of nine-point stencils with five-point stencils. In [4] it is shown that, when A comes about by a standard discretisation of a Poisson equation in two dimensions on a rectangular, uniform grid, the CPU-time for CG with RRB-preconditioning is not more than $O(N^{9/8})$. In [1] it is shown that an optimal order of convergence can be obtained by using such an incomplete decomposition and certain combinations of V-cycles and W-cycles. In [11] a different lumping procedure is used (e.g. a nine-point stencil is reduced to a five-point stencil using an approximation based on a linear interpolation) and the incomplete decomposition is combined with a multigrid method.

We have followed a different approach: after a renumbering of the unknowns according to the partitioning based on multigrid, L and U are obtained from a modified incomplete LU-decomposition based on a drop tolerance. In the sequel of this paper, this technique is referred to as Nested Grids ILU-decomposition (NGILU). This decomposition includes the modification of the main diagonal as described in [8]. The renumbering is similar to that of ILUM-factorization described in [12] in which the following idea is applied recursively: form the reduced system and apply a dropping strategy to this system. However, in the approach we follow, the renumbering is applied only once on the whole system of linear equations, and not each time when a reduced system has been formed.

In Section 2 the partitioning of the unknowns and the construction of the factors L and U is described. Some numerical results are given in Section 3. In Section 4, some conclusions are drawn, and some suggestions for future research are made.

2. The preconditioning technique

First we describe the renumbering of the unknowns. Consider a sequence of nested grids $\Omega_1, \Omega_2, \dots, \Omega_\gamma$, where $\Omega_\gamma \subset \Omega_{\gamma-1} \subset \dots \subset \Omega_1$. If all grids are uniform, Ω_m has mesh size $2^{m-1}h$, where h is the mesh size of the finest grid Ω_1 . The set of unknowns at the m th level is now defined by $W_m = \Omega_m \setminus \Omega_{m+1}$, where $\Omega_{\gamma+1} = \emptyset$. If the numbering within the levels is lexicographical, and if Dirichlet boundary conditions are used, we obtain for the inner grid points of a rectangular 8×8 -grid with constant mesh size:

| | | | | | |
|----|----|----|-----------|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 |
| 7 | 28 | 8 | 29 | 9 | 30 |
| 10 | 11 | 12 | 13 | 14 | 15 |
| 16 | 31 | 17 | 36 | 18 | 32 |
| 19 | 20 | 21 | 22 | 23 | 24 |
| 25 | 33 | 26 | 34 | 27 | 35 |

The points with numbers 1 to 27 belong to the first level W_1 . Similarly, the two sets of points 28 to 35 and **36** belong to W_2 and W_3 , respectively (note that the number of inner grid points in one direction does not necessarily have to be a power of 2). In [15] an algorithm for the generation of such a numbering is given in the more general case where the mesh is not uniform. This algorithm is based on a sequence of uniform grids which are laid over the physical domain determining to which level the points belong. The idea is similar to that of adaptive multigrid [9]. Numbering the unknowns as described above results in a system of linear equations which can be written as

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix},$$

where x_1 is the vector containing the unknowns of the first level W_1 , and x_2 those of the second grid Ω_2 . The partitioning of the matrix can be repeated for the matrix in the lower-right corner until we arrive at the coarsest grid. In order to obtain factors L and U which enable us to make an efficient implementation of the statement $y := (LU)^{-1}z$ on supercomputers, it is advantageous to use an appropriate choice for the ordering of unknowns within each level. For example, one can use a red-black ordering.

2.1. Construction of the factors L and U

The preconditioning technique consists now of making a splitting $A = LU + R$. In [14], several methods for choosing such a splitting were compared to each other, and it appeared that strategies based on a threshold parameter led to the best results. Therefore, we construct the splitting in such a way that the elements of the residual matrix r_{ij} all satisfy $|r_{ij}| \leq \varepsilon_{ij}$. This is realized by making an incomplete LU-decomposition of A based on the threshold parameter ε_{ij} in a similar way as in [13]. First the main diagonal of A is scaled to unity. Next, the incomplete decomposition is

constructed row by row. Given the first $i - 1$ rows of L and U , we construct row i of L and U from $R = A - LU$ as

$$r_{ik} = a_{ik} - \sum_{j=1}^{\min(i,k)} l_{ij} u_{jk}. \quad (1)$$

Suppose l_{ij} has been calculated for $j < k$. If $k < i$ one obtains from (1)

$$r_{ik} + l_{ik} u_{kk} = a_{ik} - \sum_{j=1}^{k-1} l_{ij} u_{jk}, \quad k < i. \quad (2)$$

If the absolute value of the right-hand side of this equation is less than ε_{ij} , fill-in on position (i, k) is added on the main diagonal (Gustafsson's modification [8]: by this choice a constant vector is in the kernel of R); otherwise l_{ik} is calculated from (2) together with $r_{ik} = 0$. With $l_{ii} = 1$, u_{ii} can be calculated from

$$u_{ii} = a_{ii} - \sum_{j=1}^{i-1} l_{ij} u_{ji}. \quad (3)$$

If $k > i$, u_{ik} can be calculated in a similar way. In the special case of a positive definite, symmetric matrix A , we demand that $U = L^T$, and we can construct an incomplete Choleski-decomposition in a similar way.

2.2. Choice of the drop tolerance

In the special case where all mesh sizes are approximately the same, ε_{ij} is chosen as follows:

$$\varepsilon_{ij} = c^{m-1} \varepsilon^{(1)}, \quad \text{where } \max(i, j) \in W_m. \quad (4)$$

Starting with $\varepsilon^{(1)}$ in the first diagonal block, corresponding with level W_1 , we let the drop tolerance decrease by multiplying with a positive factor $c < 1$ at each new level, as shown in Fig. 1.

This particular choice of the drop tolerance causes an increase of the fill-in per level, which is essential for the quality of the incomplete decomposition. For most problems, the choice of the parameters $\varepsilon^{(1)}$ and c is not very critical. In 2D-problems $c = 0.2$ is a reasonable choice, but in 3D-problems the optimal value for c is smaller. This choice will be motivated by the analysis below.

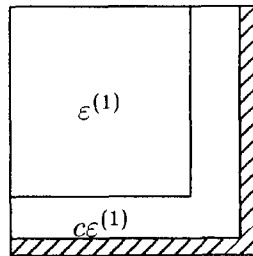


Fig. 1. The drop tolerance in case of a (nearly) uniform grid.

Though the approach is heuristic, more insight in the behaviour of the algorithm is obtained in the course of this derivation. Assume that the system of linear equations $Ax = b$ arises from some discretisation of a Poisson equation on a rectangular grid, in such a way that A is symmetric positive definite.

The criterion for choosing the drop tolerance is based on the following lemma.

Lemma 1. *Let the incomplete Choleski-decomposition of the nonsingular matrix A be given by $LL^T = A - R$. If both R and $A - \alpha R$ are positive semidefinite for $\alpha > 1$, then the condition number of $L^{-1}AL^{-T}$ is bounded by $\alpha/(\alpha - 1)$.*

Remark. Suppose that A is an M -matrix, and L results from a modified incomplete decomposition of A as described in [8]. In that case R has nonpositive elements outside the main diagonal and row sums zero. Hence from Gerschgorin's theorem it follows that R is positive semidefinite.

Proof of Lemma 1. From $LL^T = A - R$ it follows that $L^{-1}AL^{-T} = I + L^{-1}RL^{-T}$, and since R is positive semidefinite, it follows that none of the eigenvalues of $L^{-1}AL^{-T}$ is less than 1.

From $A - \alpha R$ being positive semidefinite and $R = A - LL^T$, it follows that $(1 - \alpha)L^{-1}AL^{-T} + \alpha I$ is positive semidefinite. This completes the proof of Lemma 1. \square

In the following, we try to find a condition which assures that $A - \alpha R$ is positive semidefinite for some $\alpha > 1$. Our starting point will be the necessary condition for this requirement that

$$u_k^T(A - \alpha R)u_k \geq 0,$$

where u_k is a normalized eigenvector of A . This condition is sufficient if the eigenvectors u_k are equal or sufficiently close to the eigenvectors of $A - \alpha R$. As we need indices below for other purposes, we denote u_k by q , and rewrite our condition in the form

$$\alpha q^T R q \leq \lambda, \quad (5)$$

where λ is the eigenvalue of A corresponding to q . Note that (5) is a strong requirement for low-frequency components. For example, for a nearly constant eigenvector with a very small eigenvalue λ , we see that Gustafsson's modification is necessary. Numerical experiments showed that R is of near block-diagonal form. Therefore, we restrict our analysis to the strict block-diagonal form

$$R = \begin{bmatrix} R_{11} & & & \\ & R_{22} & & \\ & & \ddots & \\ & & & R_{\gamma\gamma} \end{bmatrix}. \quad (6)$$

Consider $q^T R q$. From (6) it follows that

$$q^T R q = \sum_{i=1}^{\gamma} q_i^T R_{ii} q_i,$$

where, using the same partitioning, $q = [q_1, q_2, \dots, q_\gamma]^T$. Next suppose that

$$q_i^T R_{ii} q_i \leq \mu_i q_i^T q_i,$$

then

$$q^T R q = \sum_{i=1}^{\gamma} q_i^T R_{ii} q_i \leq \sum_{i=1}^{\gamma} \mu_i q_i^T q_i.$$

Now, for all i it holds that q_i is a restriction of q to a coarser grid. For a smooth eigenvector q , one can find a smooth eigenfunction $u(x, y)$ of the partial differential equation, such that the components of the vector q are given by the value of $u(x, y)$ in the grid points. In that case, both $q^T q$ and $q_i^T q_i$ represent, up to a factor, an approximation to the integral

$$\int_0^1 \int_0^1 u(x, y) dx dy.$$

The ratio between these two approximations can be found by substituting a constant in the respective integration rules, thus it is approximately equal to the ratio of the lengths of the vectors q and q_i . Hence $q_i^T q_i \approx 2^{-i}$ (e.g. q_1 has about half the length of q). Furthermore, suppose that there exists a real β such that $\mu_i = \mu_0 \beta^i$ and $q_i^T q_i \leq 2^{-i}$, then

$$q^T R q \leq \mu_0 \sum_{i=1}^{\gamma} 2^{-i} \beta^i = \frac{1}{2} \mu_0 \beta \sum_{i=1}^{\gamma} (\frac{1}{2} \beta)^{i-1} = \frac{1}{2} \mu_0 \beta \frac{1 - (\frac{1}{2} \beta)^{\gamma}}{1 - \frac{1}{2} \beta}.$$

Now if $\beta < 2$ then for all γ

$$q^T R q \leq \frac{\mu_0 \beta}{2 - \beta}.$$

Hence, under the assumptions made, condition (5) is satisfied if

$$\frac{\mu_0 \beta}{2 - \beta} < \lambda / \alpha, \quad \alpha > 1.$$

It should be noted that the condition $\beta < 2$ allows a growth of μ_i with i . Furthermore, numerical experiments showed that far enough away from the boundaries the blocks R_{ii} have a very regular structure and constant coefficients. This allows us to use Fourier analysis in order to obtain μ_i . The use of Fourier analysis for estimating the performance of preconditioners is, for example, described in [5] and [6]. Note also that for our model problem the eigenvectors q are in fact Fourier components with a special choice for the frequency.

Hence, we approximate R_{ii} by a linear combination of elementary symmetric finite-difference stencils. In our current algorithm all contributions to R_{ii} are balanced. More precise, define $\mu_i = \sum \delta_j$ where the sum is over all stencils, then the eigenvalues δ_j satisfy the condition

$$\delta_j < \delta_{\max} = \hat{\epsilon} \lambda \quad \text{for all } j. \quad (7)$$

Hence the stencil is allowed in R_{ii} if (7) holds, otherwise it should be included in the decomposition. In the numerical experiments we observed a moderate growth of fill-in with increasing i . Hence μ_i

will grow with the number of significant stencils (with δ_j close to δ_{\max}) but this increase is not so large that $\beta = \mu_{i+1}/\mu_i \geq 2$. In the following we will make condition (7) more specific.

Suppose that A stems from a standard five-point discretisation of a Poisson equation on an $M_x \times M_y$ -grid with mesh size h and k in the horizontal and vertical direction, respectively. When the diagonal is scaled to unity, A has the five-point stencil

$$\frac{1}{2(h^2 + k^2)} \begin{bmatrix} & -h^2 & \\ -k^2 & 2(h^2 + k^2) & -k^2 \\ & -h^2 & \end{bmatrix}.$$

The eigenvalues for a Fourier component $\exp(i(f_1 x + f_2 y))$ of this stencil are

$$\begin{aligned} \lambda_A(f_1, f_2) &= \frac{1}{2(h^2 + k^2)} [2(h^2 + k^2) - 2k^2 \cos f_1 h - 2h^2 \cos f_2 k] \\ &= \frac{2}{(h^2 + k^2)} [k^2 \sin^2(\tfrac{1}{2}f_1 h) + h^2 \sin^2(\tfrac{1}{2}f_2 k)]. \end{aligned} \quad (8)$$

Here $|f_1 h|, |f_2 k| < \pi$. Suppose that R_{ii} is the sum of stencils of the form

$$\varepsilon \begin{bmatrix} -1 & & -1 \\ & 4 & \\ -1 & & -1 \end{bmatrix},$$

with mesh sizes nh and mk in the horizontal and vertical direction, respectively. Here, n or m may be zero! Note that we may take the sum of the coefficients equal to zero, because the row sums of R are zero. The eigenvalue δ_j of this stencil is given by

$$\begin{aligned} \delta_j(f_1, f_2) &= 4\varepsilon[1 - \cos f_1 nh \cos f_2 mk] \\ &= 8\varepsilon[\sin^2(\tfrac{1}{2}n f_1 h) + \sin^2(\tfrac{1}{2}m f_2 k) - 2 \sin^2(\tfrac{1}{2}n f_1 h) + \sin^2(\tfrac{1}{2}m f_2 k)]. \end{aligned} \quad (9)$$

Now $\delta_j(f_1, f_2)$ can be bounded from above by

$$\begin{aligned} \delta_j(f_1, f_2) &\leq 8\varepsilon[\sin^2(\tfrac{1}{2}n f_1 h) + \sin^2(\tfrac{1}{2}m f_2 k)] \\ &\leq 8\varepsilon[n^2 \sin^2(\tfrac{1}{2}f_1 h) + m^2 \sin^2(\tfrac{1}{2}f_2 k)]. \end{aligned} \quad (10)$$

Note that in the first step equality occurs only if $n f_1 h$ or $m f_2 k$ is a multiple of π , and in the second step if $f_1 = f_2 = 0$. The second step holds only because of the limitations put on $f_1 h, f_2 k$ given below Eq. (8).

Condition (7) is, in this specific case, $\delta_j(f_1, f_2) < \hat{\varepsilon}\lambda_A(f_1, f_2)$. This is certainly true if it holds for δ_j replaced by the upper bound (10), hence if

$$\varepsilon < \frac{\hat{\varepsilon}(k^2 \sin^2(\frac{1}{2}f_1 h) + h^2 \sin^2(\frac{1}{2}f_2 k))}{4(h^2 + k^2)(n^2 \sin^2(\frac{1}{2}f_1 h) + m^2 \sin^2(\frac{1}{2}f_2 k))}.$$

In its turn this condition is certainly fulfilled if

$$\varepsilon < \frac{\hat{\varepsilon}h^2k^2}{4(h^2 + k^2)} \min\left(\frac{1}{(nh)^2}, \frac{1}{(mk)^2}\right).$$

We obtained slightly better results with the following choice:

$$\varepsilon < \frac{\hat{\varepsilon}h^2k^2}{4(h^2 + k^2)(n^2h^2 + m^2k^2)}.$$

Note that in the special case where the grid is uniform, $h = k$ and $n = m$ and we obtain a drop tolerance as shown in Fig. 1 with $c = \frac{1}{4}$. In our implementation with varying meshes we use

$$\varepsilon_{ij} = \varepsilon \frac{h_{ij}^2 k_{ij}^2}{(h_{ij}^2 + k_{ij}^2) \rho_{ij}^2}. \quad (11)$$

Herein h_{ij} and k_{ij} are the minimum of the mesh sizes in, respectively, horizontal and vertical direction at points i and j , ρ_{ij} is the distance between the two grid points with numbers i and j , and ε is a parameter which has to be chosen in advance.

3. Numerical experiments

In this section, we demonstrate NGILU-preconditioning by showing the results of three examples. When a lexicographical ordering of the unknowns is used, and the sparsity pattern of $L + U$ is the same as that of A , this is indicated as *standard* (M)ILU. When the sparsity pattern of the factors L and U is based on a threshold parameter ε , this is indicated as (M)ILU(ε). As stopping criterion we used

$$\|L^{-1}(b - AU^{-1}\tilde{x}^{(m)})\|_2 < 10^{-6} \|L^{-1}(b - AU^{-1}\tilde{x}^{(0)})\|_2,$$

where $\tilde{x}^{(n)} = UX^{(n)}$.

Example 1. The first example shows the results of a Nested Grids Incomplete Choleski-decomposition (NGIC) applied to a matrix which can be represented by a nine-point stencil. Consider the Poisson equation on the unit square $[0, 1] \times [0, 1]$ with condition $\partial u / \partial n = 0$ on every boundary. This problem is of interest for unsteady incompressible Navier–Stokes solvers, where at each time step the pressure has to be computed. The Poisson equation is discretised over a rectangular $(M - 1) \times (M - 1)$ -grid with constant mesh size, using a standard five-point stencil. Within the

separate levels we use a red–black ordering. Hence after renumbering and scaling we obtain the following system of linear equations

$$\begin{pmatrix} I & C \\ C^T & I \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix},$$

which can be solved by first solving the reduced system $(I - C^T C)x_2 = b_2 - C^T b_1$ and then calculating $x_1 = b_1 - Cx_2$. The Schur-complement $(I - C^T C)$ can be represented by a nine-point stencil (see, e.g., [7]). Table 1 shows the results of NGIC-decomposition combined with the Conjugate Gradient method (NGICCG) applied to the reduced system. For the drop tolerance we used (4) with $\varepsilon^{(1)} = c = 0.2$. The second row shows that the number of CG-iterations does not increase with mesh-refinement. The third row shows that the number of non-zero elements in L divided by M^2 hardly increases. As a consequence, the number of floating point operations is almost a constant times the number of degrees of freedom.

Example 2. This test problem is a simplified aquifer problem which is taken from [18]. The nonsymmetric system of linear equations comes about after the discretisation of the steady convection–diffusion equation

$$-\frac{\partial}{\partial x} \left(A \frac{\partial}{\partial x} u \right) - \frac{\partial}{\partial y} \left(A \frac{\partial}{\partial y} u \right) + B \frac{\partial}{\partial x} u = F$$

on the square $[0, 1] \times [0, 1]$. The function $B(x, y)$ is equal to $2e^{2(x^2 + y^2)}$, and the diffusion coefficient $A(x, y)$ is strongly discontinuous as shown in Fig. 2. The dashed area indicates the region in which $A(x, y) = 10000$. The right-hand side $F(x, y)$ is zero everywhere, except for the small square in the center, where $F(x, y) = 100$.

As a preconditioning technique we used an NGILU-decomposition with the drop tolerance as in (4) with $\varepsilon^{(1)} = c = 0.2$. Fig. 3 shows that the convergence rate of Bi-CGSTAB is excellent, even when the coefficients in the PDE are strongly discontinuous. From the results of Fig. 4 it appears that the convergence behaviour is relatively smooth, which is advantageous to the construction of stopping criteria when the linear solver is used as an inner-iteration method, for example, within some Newton method.

Example 3. The convergence rate of multigrid algorithms based on point relaxation smoothers deteriorates for problems with strong anisotropies (see, e.g., [3]). Anisotropic discrete operators arise, for example, in problems in which the differential operator is discretised on highly stretched

Table 1
Results of NGICCG on the reduced system

| M | 65 | 129 | 257 | 513 | 1025 |
|------------------------|-----|-----|-----|-----|------|
| # iterations | 7 | 7 | 7 | 7 | 7 |
| # nonz./ M^2 for it. | 3.6 | 4.0 | 4.3 | 4.5 | 4.6 |
| # flops/ M^2 for it. | 204 | 217 | 226 | 233 | 236 |

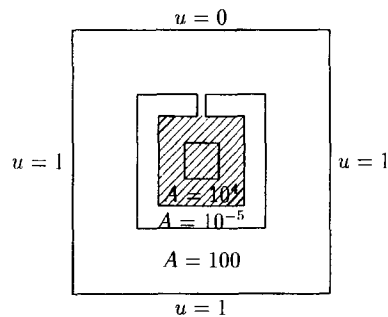


Fig. 2. The diffusion coefficient for Example 2.

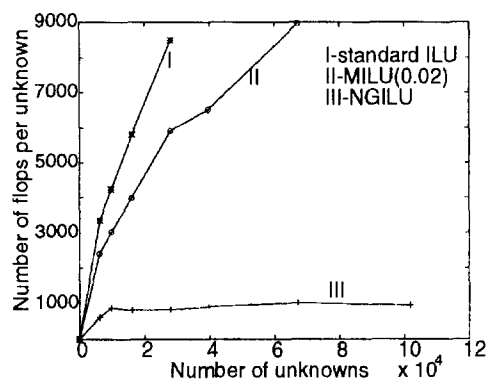
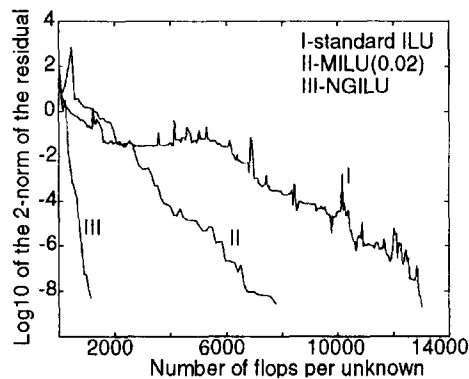


Fig. 3. Computational effort vs. grid refinement, Example 2.

Fig. 4. Convergence behaviour on a 201×201 -grid.

grids. Therefore, we tested the convergence of NGICCG for the system of linear equations which arises after the discretisation of a Poisson equation on a highly stretched rectangular grid. We took the same test problem as in [3]. We consider $-\Delta u(x, y) = f(x, y)$ on the unit square with the boundary condition $\partial u / \partial n = 0$ along $x = 0$ and $y = 0$, and Dirichlet boundary conditions along

Table 2
Numerical results for Example 3

| h_{\max}/h_{\min} | $M = 32$ | $M = 64$ | $M = 128$ | $M = 256$ |
|---------------------|----------|----------|-----------|-----------|
| 10 | 6 (9.5) | 6 (9.7) | 7 (9.7) | 7 (9.8) |
| 100 | 5 (10.8) | 5 (11.5) | 7 (11.2) | 9 (11.2) |
| 10^3 | 4 (12.1) | 5 (16.3) | 6 (15.7) | 10 (15.1) |
| 10^4 | 3 (11.6) | 4 (15.6) | 5 (22.0) | 9 (25.1) |
| 10^5 | 3 (11.0) | 3 (15.3) | 4 (21.7) | 7 (28.4) |
| 10^6 | 3 (10.8) | 3 (15.0) | 5 (22.4) | 6 (31.0) |

the other boundaries. An exponential stretching of an $(M + 1) \times (M + 1)$ -grid was used in both coordinate directions. The stretching was done in such a way that the minimum mesh sizes occur near the boundaries with Neumann boundary conditions, and the largest mesh sizes occur near the other boundaries. Table 2 lists the number of iteration steps of CG for various choices of M and h_{\max}/h_{\min} . For the drop tolerance ε_{ij} we used (11) with $\varepsilon = 0.2$. The average number of entries in one row of L is given between brackets. When a strong refinement of the grid is used, the factor L contains more elements, but the number of CG iteration steps combined with the resulting preconditioner remains very small.

4. Conclusions

In this paper, we have described a preconditioning technique (NGILU) which uses a partitioning of the unknowns based on a similar sequence of grids as in multigrid. Renumbering the unknowns according to this partitioning enables us to construct an incomplete LU-decomposition which can be used in eliminating effectively both high- and low-frequency errors. The factors L and U are obtained from an incomplete decomposition based on a drop tolerance. Results are presented from the above method applied to test problems including strongly varying coefficients, advective terms and grid refinement. In all cases, the method is much cheaper than standard (M)ILU. This difference is more pronounced for the really difficult problems and increases with the dimension. From numerical experiments we observed that, for many other problems, an NGILU-decomposition is superior to a standard (M)ILU-preconditioning (see [15] and [16]).

In the present method, the renumbering of the unknowns is based on a rectangular grid. The authors are currently working on an algorithm which produces a renumbering using only the matrix entries. This variant enables us to use the NGILU concept over a wide range of problems: for example, for systems of linear equations that arise after discretisation of PDEs on irregular domains using some finite element package. The first results of this method look very promising.

Acknowledgements

The authors wish to thank Prof. Dr. Henk A. van der Vorst for providing the FORTRAN-code for the second test problem, and the referees for many useful comments.

References

- [1] O. Axelsson and V. Eijkhout, The nested recursive two-level factorization method for nine-point difference matrices, *SIAM J. Sci. Statist. Comput.* **12** (1991) 1373–1400.
- [2] O. Axelsson and G. Lindskog, On the eigenvalue distribution of a class of preconditioning methods, *Numer. Math.* **48** (1986) 479–498.
- [3] E.F.F. Botta and F.W. Wubs, The convergence behaviour of iterative methods on severely stretched grids, *Int. J. Numer. Methods Eng.* **36** (1993) 3333–3350.
- [4] Cl.W. Brand, An incomplete-factorization preconditioning using red-black ordering, *Numer. Math.* **61** (1992) 433–454.
- [5] T.F. Chan and H.C. Elman, Fourier analysis of iterative methods for elliptic problems, *SIAM Rev.* **31** (1989) 20–49.
- [6] J.M. Donato and T.F. Chan, Fourier analysis of incomplete factorization preconditioners for three-dimensional anisotropic problems, *SIAM J. Sci. Statist. Comput.* **13** (1992) 319–338.
- [7] H.C. Elman and G.H. Golub, Iterative methods for cyclically reduced non-self-adjoint linear systems, *Math. Comput.* **54** (1990) 671–700.
- [8] I. Gustafsson, A class of 1st order factorization methods, *BIT* **18** (1978) 142–156.
- [9] S. McCormick and J. Thomas, The fast adaptive composite grid (FAC) method for elliptic equations, *Math. Comput.* **46** (1986) 439–456.
- [10] J.A. Meijerink and H.A. van der Vorst, An iterative solution method for linear systems of which the coefficient matrix is a symmetric M -matrix, *Math. Comput.* **31** (1977) 148–162.
- [11] A Reusken, Multigrid with matrix-dependent transfer operators for convection-diffusion problems, Tech. Report RANA 93-10, Department of Mathematics and Computing Science, Eindhoven Univ. of Technology, 1993.
- [12] Y. Saad, ILUM: a parallel multi-elimination ILU preconditioner for general sparse matrices, Tech. Report UMSI 92-241, Univ. of Minnesota Supercomputer Institute, 1992.
- [13] Y. Saad, ILUT: A dual threshold incomplete LU factorization, Tech. Report UMSI 92-38, Univ. of Minnesota Supercomputer Institute, 1992.
- [14] A. van der Ploeg, Preconditioning techniques for non-symmetric matrices with application to temperature calculation of cooled concrete, *Int. J. Numer. Methods Eng.* **35** (1992) 1311–1328.
- [15] A. van der Ploeg, Preconditioning for sparse matrices with applications, Ph.D. Thesis, Univ. of Groningen, 1994.
- [16] A. van der Ploeg, E.F.F. Botta and F.W. Wubs, Grid-independent convergence based on preconditioning techniques, Tech. Report W-9310, Department of Mathematics, Groningen, 1993.
- [17] A. van der Sluis and H.A. van der Vorst, The rate of convergence of conjugate gradients, *Numer. Math.* **48** (1986) 543–560.
- [18] H.A. van der Vorst, Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems, *SIAM J. Sci. Statist. Comput.* **13** (1992) 631–644.